The Boston Computer Society

BCS

SINCLAIR-TIMEX USER GROUP NEWSLETTER

Volume 2, Issue 6 August 1984

LATEST NEWS

The Aug. meeting was quite lively. Don Weiss, our librarian, exhibited his word-processing system, which begins with a TS1500 + 16K, using a Compusa disk drive for mass storage, feeding a Seikosha 150 printer driven by a Memotech Centronics parallel interface. He also has seperate keyboard with a space-bar. Unfortunately a system bug prevented his demonstrating WORD-SINC, a popular program, but he was able to show other operations.

Marty Warner, from Games to Learn By, demonstrated MSCRIPT, the wordprocessing program which was to have been available for the TS2068 along with a Mannesmann/Talley printer which was also planned as a TS product. Marty has alsoe tried TASWORD II and discussed pros and cons of both these superior programs. She offered a copy of MSCRIPT as a door prize, which was won by John Kemeny.

Then, as an encore, she explained how to make internal color adjustments on the TS2068. Watch for details in our next issue.

Finally, "Things move fast in the computer industry." Our director, Sue Mahoney, brought copies of the Connecticut Computer Society News , which among other items of interest, announced her engagement to Todd Tolhurst, after a brief courtship conducted largely over Compuserve. Details next issue.

The Absolute Beginner's Guide to Machine Code Programming on the ZX81/TS1000/TS1500

by Jack Hodgson

Over the years I've been amazed at the number of beginner programmers who are interested in machine language programming. Others might say that a beginner should avoid this type of stuff because it's far too complicated for them but I don't agree.

Con. p.9

DIRECTOR'S COLUMN

Susan Mahoney

News from Sinclair . . .

Maggie Bruzelius, Executive VP of the U.S. branch of Sinclair Research, Ltd., resigned on July 20, 1984 to take a position as marketing director for Alpha Software of Burlington, MA.

The search for a replacement is being handled personally by Nigel Searle, Managing Director of Sinclair Research, Ltd., Cambridge, England.

Over the years, our group has always enjoyed and appreciated the support of Sinclair Research (Clive Sinclair is a patron member of BCS). Best of luck to you Maggie, and thanks for all of your help.

* * *

Sinclair's QL (Quantum Leap), based on Motorola's 32-bit 68008 microprocessor chip, is still slated for release in the U.S. this November. However, since production of U.S.-bound QL's will not peak until the first quarter of 1985, initial quantities will be limited.

If you're interested in purchasing a QL, Sinclair Research will be happy to take your name and mail you an announcement when they are ready to take orders. Write to:

Sinclair Research, Ltd.
50 Stanford Street
Suite 800
Boston, MA 02114

con. Pg.8

M E R G E " "

With any luck, most of you have received the late April issue of this newsletter. It seems the Postal Service misplaced a bag or two. We're planning an extra issue this fall anyway, for the upcoming East Coast Computer Faire, which will be part of our 4th "Birthday" celebration this Fall Your suggestions and contributions will be appreciated.

WJS

MEETINGS

The July main meeting was attended. Mike Coughlin explained the nature of T.V. sets and their idiosyncracies when hooked up to a computer. Future activities were discussed and programs proposed for future meetings. The August MC meeting at Mitre featured as promised a presentation by Dave Wood on the secrets of the ZX81's internal juggling act, whereby it concurrently processes up to three signals at once. Just what's inside the NOP generator anyway ? Future meetings probably won't be so intense; all those interested in Machine Code at any level are encouraged to attend. Next meeting:

September 6, 1984

CALL FOR ADVERTISERS

The next three issues, Sept., Celebration, and Oct. will offer an excellent opportunity to reach an interested and active section of the surviving TIMEX/Sinclair community. Contact either Sue Mahoney, Group Director or the editor for special rates for all three of these issues. Members wishing to sell off equipment, offer services or software are also reminded that classified ads are available at economy rates.

This newsletter is produced by the Sinclair/Timex User Group of the Boston Computer Society in addition to regular meetings and activities. It is mailed free to members and supporters; back issues are available for a small fee.

DIRECTIONS TO MEETING

We meet in the Large Science Auditorium (Room 8/2/009) on the Harbor Campus of U/Mass Boston, which is located 3 miles from downtown Boston just off Morrisey Blvd. Follow signs from the SouthEast Expressway, Exit 17. During reconstruction, Columbia Road, which comes in from Dorchester may be a better connection.

On the MBTA, take the Red Line (Ashmont) to Columbia Sta. Use the free University Shuttlebus from the T parking lot to get to campus.

The Sinclair TIMEX User Group

Sue Mahoney Director
c/o BCS Office
or 203-755-2699

Will Stackman Editor
210B Summer St.
Somerville MA 02143
617-666-8626

Jack Hodgson Correspondent
P.O.Box 526
Cambridge MA 02238
617-354-7899

John Kemeny
User Group Correspondent
284 Great Rd. Apt. D5
Acton MA 01720

Allan Cohen
Meeting Coordinator
617-961-3453

LIBRARIAN
Don Weiss
27 Mitchell St.
Randolph MA 02368
(617) 986-8449

THEY'RE OUT THERE!

The newsletter exchange continues provide us with the doings of TS Users across the continent. Triangle Sinclair User's Group is in hot pursuit of the 64 column screen and has published the detailks of an RGB hook-up designed by John Oliger of Cumberland, IN. Doug Dewey, their fearless leader is ready to market his Chameleon for $60. Call 919-929-3079 for details

Want an audible keybeep ? ZX World News Bulletin suggests POKE 23609, (#from 1 to 255). What's your favorite ? They are publishing programs from an Italian correspondant, Oriani Massimo.

Fred Nachbaur's **Syncware News** is now coming out under the aegis of our northern friend, Tom Woods. The magazines' new address; P.O.Box 64, Jefferson, N.H. Nov.-Apr. (delayed issue) had instructions for using a Votem to make a cassette controller, a "Caveman" BASIC Wordprocessor for both TS100 and TS2068 (the differences are instructive), a page of schematics for small loading aids, plus some high-powered math articles. May-June, Fred's last has an improved monitor driver, more math, and a Data Acquisition Development program for using the Votem or another V-F circuit board with the 1000.

TUG from Gainesville, Fla. is getting into the Westridge Modem and will be on line to talk. Roger Hunsiker of the group suggests that green PAPER and BORDER with black INK makes the most readable B&W screen for the 2068. Any comments ?

David Hosher, from ATSU in Central Ohio, gets into the colored cursor on the 2068 and uses it to highlight listings (how about red REM statements ?) To activate a colored cursor, go to E mode, then shift the color. If you shift 8 or 9, which are "colorless" you can create flashing statements without using the FLASH 0/1 commands;9 for on, 8 for off.

We continue to receive issues fron the T/S User's of Vancouver. June/July reviews Z-Speak, a simple allophone speech synthesizer and offers a short program to compute your pulse rate.

From issues early this year, which are beginning to swamp our faithful correspondent, John Kemeny, (more about which later) comes headline sized printing from New Brunswick.

FROM THE SINCLAIR PRINTER MANUAL
A PROGRAM TO ENLARGE CHARACTERS
DOESN/T SEEM TO WORK FOR TS1000

```
 10 DIM A$(40)
 20 INPUT A$
 30 FOR Y=43 TO 4 STEP -1
 40 FOR X=0 TO 7
 50 LET S=43-7
 60 LET N=X+8*INT (S/8)
 70 LET SCAN=S-8*INT (S/8)
 80 LET P=PEEK (7680+8*CODE A$(
N+1)+SCAN)
 90 FOR I=7 TO 0 STEP -1
100 PLOT 8*X+7-I,Y
110 UNPLOT 8*X+7-I,Y
120 IF P<2**I THEN GOTO 150
130 LET P=P-2**I
140 PLOT 8*X+7-I,Y
150 NEXT I
160 NEXT X
170 NEXT Y
```

Unfortunately, this doesn't seem work on the 2040 printer. Anyone see a fix ?

Finally, from TIMELINEZ, the newsletter of the Triangle Timex ser's Group of N.C. a method for darkening the display file image on the screen which will also COPY or LIST to the printer. See lines 1 through 9.

The rest of program is derived from several British sources. For an explanation of this "toy" see the last few issues of Creative Computing, the Math and Art series.

There is a wealth of material out there. We need help mining it for the newsletter and for your use. Volunteers are needed to compile an index. TS2068 users especially should want to participate in such a project. Contact John Kemeny for more information.

WJS

On the Magazine Scene

One of the most interesting magazines on the news stands in Harvard Square is a British publication, **Your Computer**, which covers all the smaller personal computers on the English market. Naturally the ZX81 and the Spectrum figure prominently on its pages. Among the tips published so far this year is the secret to INPUTing anywhere on the TS2068 (Spectrum) screen. Key in;

```
10 INPUT AT 22,0;AT 1,10:"Promp
 t message"; var.
```

Experienced BASINC users will recognize this as a variant on the technique for speeding up screen displays. The second screen location can be anywhere legal. So much for that gripe! Courtesy of Hubert Surrer from somewhere in Germany.

Programs range from the weird to the wonderful. The May issue, (the latest I've seen) has a software generated speech synthesizer! for the ZX81, as a follow up to an article last year on the same feature for a Spectrum, a Quickload for the latter with a top speed of over 3600 Baud, arcade games for both machines, and other goodies.

Even the ads are of some interest. Watch for the Floopy from 'Phi Mag which uses a loop of standard sized cassette tape and a nine-track recording head (one for error checking). Claims an average access time of 3 seconds, data transfer of 10K bytes per second. Will load 32K in just over 3 seconds.

Earlier issues this year contained items of interest such as "Function Keys" for the Spectrum (which would require some address changes for the TS2068) and definable printer characters for the ZX81.

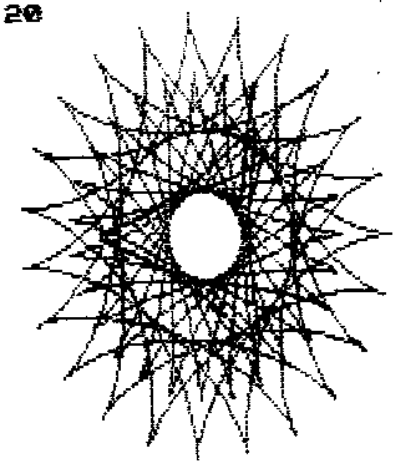
Other general interest British magazines, such as "What Micro?" support Sinclair computers, but none to the depth of "Your Computer"

WJS

```
   1 CLEAR 56575: REM set ramtop
   2 LET s=57786: REM start addr
ess
   3 READ n
   4 IF n=-1 THEN GO TO 8: REM d
ata flag
   5 POKE s,n
   6 LET s=s-1
   7 GO TO 3: REM POKE next addr
ess
   8 STOP
   9 DATA 17,0,221,213,1,0,3,42,
54,92,36,126,167,31,182,18,35,19
,13,32,246,16,244,225,37,34,54,9
2,201,-1
  10 REM * S P Y R O - G Y R O *
  20 BORDER 6: PAPER 0: INK 7
  30 LET ov=0: CLS : GO TO 150
  50 LET a=0: LET b=0: LET f=0:
LET i=.1: LET h1=0: LET v1=0
 100 LET b=-(a*L)/s: LET v=INT (
((L-s)*SIN (a)-p*SIN (b))+.5): L
ET h=INT (((L-s)*COS (a)-p*COS (
b))+.5)
 110 IF f=2 THEN PLOT 128+h1,88+
v1: DRAW h-h1,v-v1
 115 IF f<2 THEN LET f=f+1
 120 LET h1=h: LET v1=v: LET a=a
+i
 125 LET a$=INKEY$: IF a$="" THE
N GO TO 100
 130 GO TO 310
 150 PAPER 0: INK 6: CLS : PRINT
 '"          S P Y R O - G Y R O "'
"                                "
 160 INK 7: PRINT '"   This prog
ram draws patterns    which can
be generated by two   gears show
n below-"
 170 CIRCLE 60,51,40: CIRCLE 84,
51,16: INK 3: PLOT 60,51: DRAW 0
,40: PLOT 84,51: DRAW 0,16: PRIN
T AT 12,7;"L";AT 14,10;"s": PLOT
 94,51: DRAW 16,32: PLOT 94,51:
DRAW 18,30
 180 INK 6: PRINT AT 11,12;"pen"
 190 INK 7: PRINT AT 10,16;"  Th
e smaller";AT 11,16;"gear rotate
s";AT 12,16;"around the";AT 13,1
6;"inside of the";AT 14,16;"larg
er gear,"
 200 PRINT AT 15,16;"producing a
 pen";AT 16,16;"trace."
 210 PRINT AT 17,16;"  Input L,s
,and";AT 18,16;"pen position";AT
 19,16;"in small circle."
 220 PRINT AT 21,11;"Hit a key":
 PAUSE 0
 225 INPUT " Load a Pattern from
 Tape? y/n";x$: IF x$="y" THEN I
NPUT "Filename?";f$: CLS : LOAD
f$CODE : PAUSE 0: GO TO 150
 227 PAPER 7: INK 0
 230 INPUT "Radius L (0-85)?";L:
 PRINT L;" ";
 240 INPUT "Radius s ?";s: PRINT
 s;" ";
 250 INPUT "Pen position inside
B ?";p: PRINT p
 255 INPUT "Inkcolor ?";ic: INK
ic
 256 INPUT "Papercolor ?";pc
 300 PAUSE 50: IF ov=0 THEN PAPE
R pc: CLS : GO TO 50
 305 GO TO 50
 310 INPUT "Overprint another pa
ttern ? y/n";q$: IF q$="y" THEN
LET ov=1: GO TO 230
 315 IF q$="y" THEN GO TO 30
 320 INPUT "Print Hardcopy ? y/n
";q$: LPRINT L;" ";s;" ";p: IF q
$="y" THEN COPY
 330 INPUT "Save Tape copy ? y/n
";q$: IF q$="y" THEN INPUT "File
name ?";q$: SAVE q$SCREEN$
 340 GO TO 150
```

RANDOMIZE USR 57786 to darken
POKE 23607,60 to restore

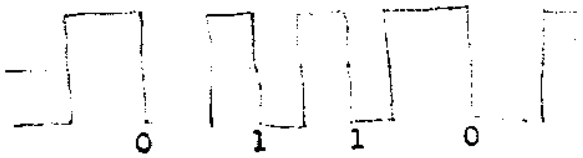
```
85 30 20
60 30 12
85 30 20
```



A Comparator to Use on Bad 2068 Tapes

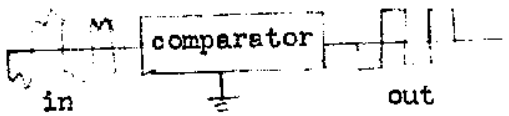
by Chuck Ludiwsky
as told to Mike Coughlin

The program loading and saving process on the TS2068 is a great improvement over the TS1000. The signal recorded is a series of square pulses, with ones



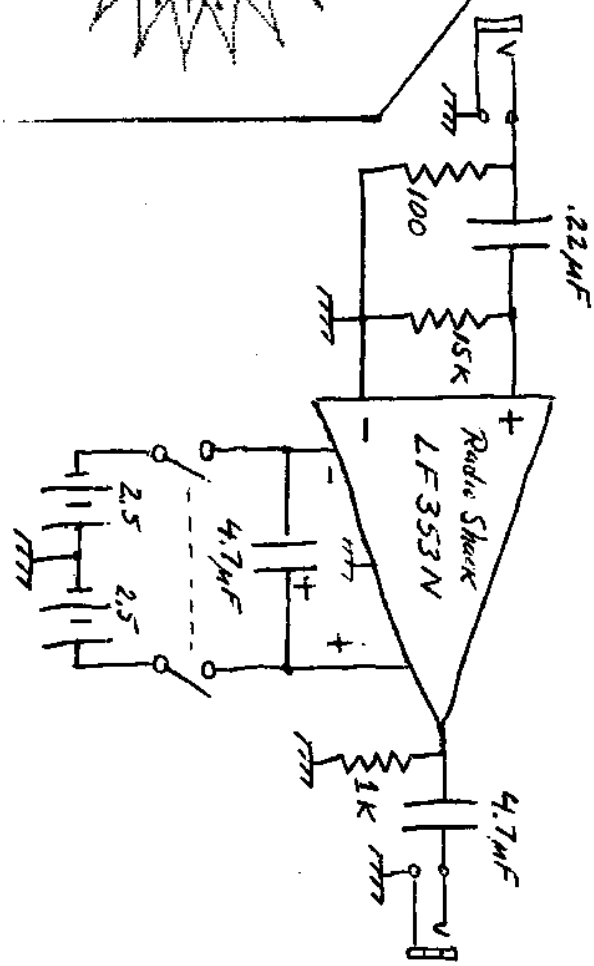
shorter than zeros. There is no DC component and reversing plus and minus has no important effect. Square waves are difficult to record and some tapes are being sold that may not load.

A tape would have to be in really dreadful condition before the point where the signal changed from plus to minus was effected. A handy electronic circuit known as a comparator



is specially designed to take a bad square wave and make it into a good square wave. An ordinary op-amp makes a good comparator for our purpose. While an LF353N was used in our example, any op-amp that will work at low voltage is suitable.

A comparator is a very high gain amplifier whose output is designed to go from one extreme of the power supply to the other. Think of it as a constantly overloaded amplifier and a one bit A to D converter. The output is equal to the power supply voltage. The values of the components are not critical, so don't be afraid to experiment.



WANNA GET SOMEBODIES' ATTENTION
Try this on your TS2068!!!!!!!!

```
991>FOR T=1 TO 5
993 FOR X=0 TO 7
995 BORDER X
997 BEEP .03,X
998 NEXT X
999 NEXT T
```

USING BOB MASTER'S BOOK

Most TS1000/1500(ZX81) owners and not a few TS2068 users have Psion Ltd.s VU-CALC and VU-FILE (The Organizer). But even those who have applications developed for these two utility programs probably do not use either as much as they might. The in-box documentation which accompanies these programs is sketchy at best, and barely scratches the surface of their potential usefulness.

"Getting Serious with Your TIMEX/SINCLAIR" could end the under-use of either program. Expanded from tutorials BOB has given at S/T Users' Group meetings, this book walks through a range of examples which can yield results of the sort we usually claim we bought a computer to get.

The appendix by Mark Fisher on VU-CALC for the 2068 will allow those of us with color computer to benefit from this book as well. Although this spreadsheet can be as large as 50x50 (Rows from A to AX, columns from 01 to 50), the templates included with this book will serve for most applications. Larger forms can be created on fullsize accounting paper.

Both versions of VU-Calc can be printed out using the 2040 Printer. In vertical format, a TS100 version can be pasted up on 8 1/2 x 11 to 6 columns by 26 rows; horizontally, 9 columns by 18 rows. (Six screens in either case). On the 2068, vertical yields 8 by 50; horizontal, 12 by 36.

TS1000 users should also consider following Fisher's customizing notes. Carefully entered, these improvements are worth adding to most uses of VU-CALC.

The whole secret to getting the most out of this book is to sit down and use it. As with most spread sheets, developing formulas is really a form of programming using simple accounting and the "language" of the program at hand.

VU-FILE (The Organizer) seems at first a simpler piece of software, but may be harder to use effectively. Bob's suggestions for formatting are ingenuous and increase the usefulness manyfold. His demystification of print formatting procedure is invaluable. Again, most explanations are clear only after use.

Anyone seriously expecting to use VU-FILE will need to study the formatting templates supplied. Otherwise you'll spend (waste) hours trying to get records the right size. Perhaps the major lesson of this book is that effective planning before using an application program, applying an old-fashioned pencil to an appropriate form will yield maximum results.

Too often, users assume the computer will do it all. There's not a little laziness in the demand for "user-friendly" software. A good tool may take a while to learn to use well.

To reiterate a point. If carefully though-out documentation were supplied with more software, it would be more often worth its price. Unfortunately, many programs are released hot off the printer. It will be interesting to see what the documentation for the software bundled with the QL is like.

WJS

A BUG IN TS2068 VU-CALC !

Option 2 on the end menu, which clears the sheet for a new model crashes the program. GOTO 9000 will not display the error. Variables have been overwritten.

Repairs are simple. Enter #q to exit to BASIC. The LIST 3200, and bring it down to EDIT. Add the following statement to the head of the line before the DIMs.

```
3200 CLEAR 29327:
```

and leave the DIMs, etc in place. Return to VU-CALC by entering GOTO 3200.

Discovered by Gary Szekeres, T/S Users Group of Cincinnati

Reserving Space in a REM Statement

When storing a machine code program in a REM statement, the programmer must have written the statement with at least as many bytes as the program requires; often a tedious and time consuming task, especially if the program is large. Assuming that the first line in BASIC is the REM statement (1 REM xxx...) the first addressable byte is at 16514 (hex4082). The following procedure will simplify loading MC into 1 REM.

Step 1 : Enter 1 REM; 10 Program loader.

Step 2 : Place this short MC program above Ramtop, at, say, 30000 (hex7530).

Address	HEX	Mnem
7530	21 82 40	LDHL 4082 Addr.1st byte
7533	01<st qr>	LDBC qrst No.reserved bytes
7536	CD 9E 09	Call 099E Make room
7539	C9	Ret

Step 3 : RAND USR 30000 (safe address above Ramtop)

Step 4 : POKE into 16511 and 16512 the number qrst + 2 (low byte into 16511, high byte into 16512)

Following Step 4, the machine code program may by entered by POKEing the code into addresses beginning at 16514, using any standard loading program.

For example, suppose the MC program uses 1000 bytes. The number qrst = 03 E8 in hex; the instruction at address 7533 is 01 E8 03 (note reversal). After Step 3, the number 1002 would be POKEd into addresses 16511 and 16512 in the immediate mode, i.e

```
POKE 16511,234  <enter>
POKE 16512,3    <enter>
```

(Note; 3 x 256 + 234 = 1002)

Contributed by Arthur Kant, 139 Woodridge Rd., Wayland MA 01778

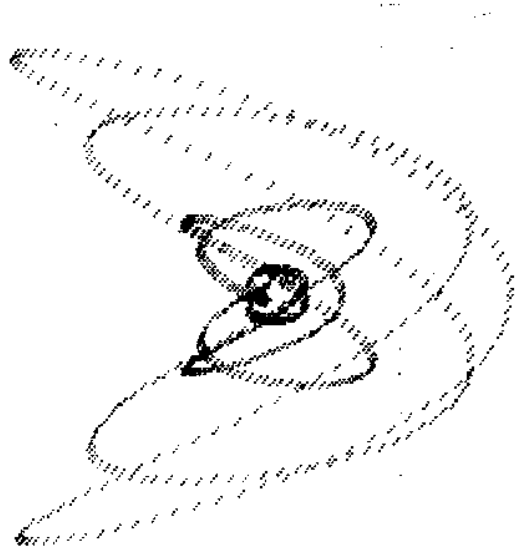
"FUN MATHEMATICS on your microcomputer"

Czes Kosniowski, from the Computer Dept. of the University of Newcastle on Tyne, has written a maths book for those who always suspected that computers probably could be used as mathematical tools, but that somehow that wouldn't be "friendly".

This moderately advanced text (at last past freshman year of highschool) contains very clear discussions of important principles which support the screen display of mathematical functions.

The accompaning BASIC programs can be run on most Microsoft-based home computers including all Sinclair/Timex models. Several have been specifically converted into BASINC, most require some knowledge of its idiosyncracies. The polar graphs which accompany this review show some of the posibilities inherent in mathematically based computer "art".

Published by Cambridge University Press; ISBN 0 521 27451 6. Available locally under $10. Get this one into your library.



R = 1+2*COS (Z) A= 9 B= 5

News from Timex . . .

Where, O, Where are the TS2068 technical manuals? I've spoken with many people who ordered the new tech manual from Little Rock in April or May and still have not received them, while several other persons have. Why should this be?

It turns out that Timex originally produced 200 copies of the approximately 300-page manual on a Xerox machine. As the orders began to outnumber the manuals, it was decided to send the manual to the printer instead.

Three months later, during the week of July 30, Little Rock mailed 1,000 tech manuals - bulk rate. So, it may be near the end of August before you find a present from Timex in your mailbox.

One final note: Timex is now putting all new orders for the manual on "hold" until they have enough to warrant a second printing. For those of you who wish to order, send your request (along with a check for $25.00) to:

Timex
P.O. Box 1378
Little Rock, AR 72203
ATTN: Material Sales Div.

And Good Luck!

* * *

Timex has sold its entire inventory of computer-related products to wholesalers and retailers. This means that you CANNOT purchase any of these items directly from Timex anymore. From this point forward, it will be very important for all of us to share information on the whereabouts of Timex computer products.

Computer Faire/East . . .

You've heard of the West Coast Computer Faire? Well, it's coming East to Boston. The producers of the three-day show (October 2-4) are providing a booth for the Sinclair/Timex User Group (all those other BCS groups get a booth too). What we need are:

- Ideas. What are we going do with this opportunity?
- Volunteers. For planning, staffing, and generally hanging around the booth.

Since the show will fall near our third anniversary, let's try and make this opportunity into a showcase and celebration for our group. Please get in touch with me if you have ideas, or if you'd like to help out in the booth.

(RETurns).

Originally, I wanted to show you, in this first installment, how to load this program into your computer and run it. But I've run out of room and you have enough to chew on anyway.

Next time I'll tell you about loading machine language programs; getting "letters" from the keyboard, and the infamous, "Hex numbers".

Machine language is not mysterious. It is a little more complicated than some "high-level" languages because in machine language you must keep track of many details yourself. But it can add to the sophistacation of your programming and speed up you routines. And finally many programmers like to simply "work closer to the machine."

If you got this far you'll love installment two.

Jack

I was only a beginner when I first plowed my way through Toni Baker's book. If I learned it, anyone who wants to can.

The biggest problem I had in learning all this was that the Z80 processor chip that's in the Sinclair is so sophisticated that it had too many instructions for me to deal with. I didn't know where to start.

What I'm going to do in this series of articles is to present a simplified view of Z80 programming. I won't show you all the instructions or make you understand all the details of how it works. I will give you enough info to write simple, useful machine language programs.

This series will deal with the ZX81/TS1000/TS1500 series of computers. Many of the priciniples I demonstrate will apply to the TS2068 as well, but some of the details will have to be adjusted. Ask me or another machine language programmer at the meetings about the differences. If there's enough interest the editor of this letter may allow me to write an updated TS2068 version.

OK let's get started.

A computer is just a collection of storage places where one puts numbers. The instructions tell the computer to move numbers from one place to another, do arithmetic with them, compare them, and do things as a result of those comparisons.

First let's look at how these storage places are set up.

The processor has a bunch of storage locations built into it. They are called "registers". There are alot more storage locations in the area we all call RAM. The registers have names that are letters: like A, B, C, H, and L. The RAM locations have names that are numbers: 0 up to 65,535.

The computer understands instructions that tell it to move a number from one location to another, for example, "move the contents of 16514 to A". That instructions is actually called "LD A, (16514)". That means load the A register with the number that currently in location 16514.

The parentheses tell the computer to load A with the contents of the "address" 16514 instead of the number 16514 itself. If we wanted to put a particular number into A we would say "LD A, 16514" without the parenthesis.

An important note here. The computer can only deal with certain sized numbers. Our second example, using 16514, would not work because 16514 is too big for the computer to handle. The computer can only handle numbers up to 255.

What follows is as complicated this is going to get. So please take a deep breath and understand it.

The number "fifteen" can't be understood by a desktop calculator when it's written that way. You need to divide it into "digits" between 0-9. We all know that "fifteen", turned into digits is "15" but understand how that happened. "15" is "1" times 10 plus "5" times 1. That's called "base ten". Computers use "base 256" (Please don't everyone holler at me. I know people always says computers use base two. They do, but you don't need to know anything about that here. Just THINK BASE 256).

Converting numbers from base ten to base 256 is easy. You look it up in a chart. For the stuff we'll be doing here I'll supply the conversions. Next installment I'll show

OVER

OK. Enough of this fooling around. Let's write a program. Here's a BASIC program:

```
10 PRINT "HELLO WORLD."
```

How would we do that in machine language? It's really not too hard.

ADRS	OP CODE	INSTRUCTION
64-130	33-144-64	LD HL, 64-144
64-133	126	LD A, (HL)
64-134	254-255	CP 255
64-136	200	RET Z
64-137	205-8-8	CALL 8-8
64-140	35	INC HL
64-141	195-130-64	JP 64-133
64-144	45-42-49-49-52-0-60-52-55-49-41-27-255	

ADRS is the RAM location where these instructions are going to be stored. OP CODE is the numbers that the computer understands as instructions.The numbers starting at 64-144 are the code numbers for the letters "HELLO WORLD."

The first instruction LD HL puts the starting address of these letters into the HL register. LD A, (HL) puts the first "letter" into A. CP 255 compares whatever's in A to see if it's equal to 255. RET Z ends the program if the comparison we just did is equal to 255 (notice that the last "letter" is 255).

CALL 8-8 tells the computer to executea routine that is already in the computer, at address 8-8. That routine causes the "letter" in the A register to be printed on the screen. INC HL causes the HL register to be increased by one so it now is the address of the next "letter". JP 64-130 causes the computer to go back to the LD A, (HL) and get the next "letter".

It does all this, getting letters and printing them, until it gets to the "letter" 255 then it stops

you how to make a chart of your own.

Anyway, where were we? Oh yah. 16514 is 64-130. That is, 16514 is "64" times 256 plus "130" times 1. LD A, 16514 is written for the computer as LD A, 64-130. LD A, (16514) is written LD A, (64-130).

Sometimes we want to store numbers bigger than 255 in a register. We can do this by combining two registers into a bigger one. Some registers are set up to be used in pairs. B&C and H&L are ready to be combined. They are then called by their combined name: BC or HL. We can put numbers in them by using LD HL, 64-130 or LD BC, 64-130.

Up above we asked to computer to load A with the number stored at adresss 64-130. We can also tell it to load A with the number stored at the address that's currently in register HL. For example:

```
LD HL, 63-130
LD A,  (HL)
```

This does the same thing as
LD A, (63-130).

Before the computer can execute these instructions we must convert them into numbers. The computer can't understand the letters LD A, 28. It only understand numbers.

The number for the instruction LD A, 28 is 62-28. The 62 means LD A and the 28 tells what to load it with. 62-255 would mean LD A, 255.

When your're loading a register pair you have to do something strange. LD HL, 64-130 becomes 33-130-64. 33 means LD HL and the number to be loaded is tacked on in reverse order. Whenever the computer is reading a two section number like that, it must be given to it in reverse order.

concl. p.8.

T/S 2068 ROM BUGS by John Kemeny

Well, the Technical Manual from Timex finally arrived, and it has lots of interesting things in it. Unfortunately, it is no longer available from Timex. Since the manual's audience is software developers and not consumers, most of the material concerns arcane machine code interfaces with the cartridge system (via LROS and AROS). Of interest to the BASIC programmer, however, is Section Six - "Known 'Bugs' and 'Work-Arounds'"; here's the list :

1 - Pressing ENTER multiple times with an invalid LOAD, SAVE, or other tape command causes a system reset.

2 - If a non-existent line number is specified for an ON ERR GOTO statement, followed by an error, the system will hang. In case you're interested in the details, "the ROM code is in an endless loop trying to report the absence of a valid error handler to the non-existent error handler!!!" That's what happens when you do recursion without defining initial conditions.

3 - Parameters to the SOUND command are not fully validated. This leaves the possibility of accidentally changing the I/O ports for reading the joystick. The remedy is to execute SOUND 7,63 to enable the joystick for input.

4 - If you respond to the SCROLL? message using multiple keys you will get strange results, e.g., Cap Shift/4 scrolls two pages, Cap Shift/2 dumps ROM data.

5 - When Cap Shift/0 is held down to delete characters in the Edit Line, it sometimes outputs the DELETE keyword instead. "It should not do this in autorepeat mode. This is especially noticeable when the input line is long."

These are all the documented bugs. Here's an extra, undocumented, bug.

6 - The UP-ARROW function ^ (or ** on the T/S 1000,) which for x^y is supposed to give you x raised to the y power, doesn't work for negative x's. That is, -5^2 gives -25 which is okay(^ is done before the unary minus). But (-5)^2 gives an error, and, as everyone knows, the answer is (-5)*(-5)=25. The problem is that the algorithm computes EXP (y * LN x) if x is not zero. LN x is undefined if x is negative. If x is zero it returns zero if y is positive; 1 if y is also zero (i.e.,0**0=1); and overflow error if y is negative.

That's a suprisingly short list of bugs for a system as complex as the 2068. Please let us know if you find more.

The Boston Computer Society

Annual membership in The Boston Computer Society includes mailings from any two user/interest groups (attendance at monthly meetings is unlimited).

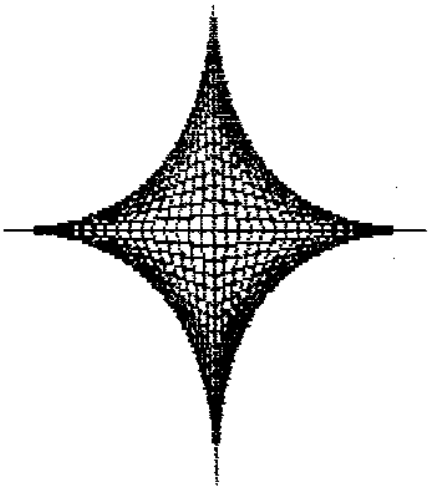
- ☐ Regular (1 year), $24
- ☐ Regular (2 year), $45
- ☐ Regular (3 year), $62
- ☐ Youth (under 19), $15
- ☐ Senior citizen, $15
- ☐ Family, $36
- ☐ Overseas, $48
- ☐ Institutional, $100
- ☐ Sustaining, $120
- ☐ Corporate, $200
- ☐ Lifetime, $2000

Sign me up for these user groups:

- ☐ Apple/Boston
- ☐ Atari
- ☐ Commodore
- ☐ Digital Equipment
- ☐ Displaywriter
- ☐ 80/Boston (TRS-80)
- ☐ Heath/Zenith
- ☐ Hewlett-Packard
- ☐ IBM PC
- ☐ Kaypro
- ☐ Macintosh
- ☐ NEC
- ☐ Osborne
- ☐ Otrona
- ■ Sinclair/Timex
- ☐ Texas Instruments
- ☐ Victor
- ☐ Other _____

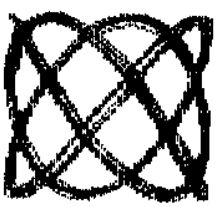
Sign me up for these special interest groups:

- ☐ Artificial Intelligence
- ☐ Business
- ☐ Consultants/Entrepreneurs
- ☐ CP/M
- ☐ Database
- ☐ Education
- ☐ Family/Home
- ☐ Graphics
- ☐ Investment
- ☐ Legal
- ☐ Logo
- ☐ Medical/Dental
- ☐ Pascal
- ☐ Real Estate
- ☐ Robotics
- ☐ Science & Engineering
- ☐ Social Impact
- ☐ Telecommunications
- ☐ Training/Documentation
- ☐ Unix & C
- ☐ Other _____



```
 10>REM ** Steve Scoville's
 15 REM        Pastime
 20 REM from TRIANGLE SINCLAIR
              USERS GROUP
 50 INPUT "Type a number ";n$
 60 IF n$="c" THEN CLS : GO TO
50
 70 LET n=VAL n$: PRINT AT 0,0;
"Input ";n;" produces :   "
100 FOR i=0 TO 85 STEP n
110 PLOT 213-i,88
120 DRAW i-85,i
130 PLOT 43+i,88
140 DRAW 85-i,i
150 PLOT 213-i,88
160 DRAW i-85,-i
170 PLOT 43+i,88
180 DRAW 85-i,-i
190 NEXT i
200 GO TO 50
```

The Boston Computer Society
One Center Plaza
Boston, MA 02108



Next Meeting: September 19, 1984